

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1294	(anchor adj text) anchortext (link\$3 adj text)	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:48
L2	4	token\$ with 1	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:39
L3	14	token\$ same 1	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:37
L4	10	3 not 2	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:37
L5	17	pars\$ with 1	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:47
L6	480	1 and (token weight score)	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:48
L7	258	1 and (weight)	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:48
L8	190	(anchor adj text) anchortext	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:48
L9	81	8 and (weight)	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 07:48
L10	18	8 and (token and weight and score)	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:13
L11	1	"6687878".pn.	USPAT	OR	ON	2006/01/25 08:09
L12	5	("6665837" "6415294" "6112203" "6633868" "6411952").PN.	USPAT	OR	ON	2006/01/25 08:09
L13	1	"6122647".pn.	USPAT	OR	ON	2006/01/25 08:09
L14	7	(US-6687878-\$ or US-6665837-\$ or US-6633868-\$ or US-6411952-\$ or US-6415294-\$ or US-6112203-\$ or US-6122647-\$).did.	USPAT	OR	ON	2006/01/25 08:09
L15	3	L14 and token\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L16	2	L14 and anchor\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09

L17	5	L14 and link\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L18	4	L14 and threshold	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L19	3	L14 and weight	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L20	0	L15 L16 L17 L18 L19	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L21	0	L15 L16 L17 L18 L19	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L22	0	L15 L16 L18 L19	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L23	0	L15 L17 L18 L19	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L24	0	L15 L18 L19	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L25	3	L14 and token\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L26	2	L14 and anchor\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L27	5	L14 and link\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L28	4	L14 and threshold	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L29	3	L14 and weight	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L30	10	hillery.xa.	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L31	46	rivette.in.	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L32	41	rivette.in.	USPAT	OR	ON	2006/01/25 08:09

L33	3	L14 and weight\$	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L34	3	((("6154213") or ("6484166") or ("6651058")).PN.	US-PGPUB; USPAT; IBM_TDB	OR	OFF	2006/01/25 08:09
L35	10	(US-6687878-\$ or US-6665837-\$ or US-6633868-\$ or US-6411952-\$ or US-6415294-\$ or US-6112203-\$ or US-6122647-\$ or US-6651058-\$ or US-6484166-\$ or US-6154213-\$).did.	USPAT	OR	ON	2006/01/25 08:09
L36	4	L35 token\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L37	4	L35 anchor\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L38	5	L35 weight\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L39	8	L35 link\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L40	6	L35 threshold\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L41	1	L36 L37 L38 L39 L40	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L42	1	L35 and (anchortext (anchor adj text))	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L43	6	L35 threshold\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L44	8	L35 link\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L45	5	L35 weight\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L46	4	L35 anchor\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L47	4	L35 token\$	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09


L48	2	L36 L37 L39 L40	US-PGPUB; USPAT; IBM_TDB	AND	ON	2006/01/25 08:09
L49	38377	((divid\$ with frequenc\$) and multipl\$)	USPAT	OR	ON	2006/01/25 08:09
L50	1	L49 and L35	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L51	51	(divid\$ near frequenc\$) and (multipl\$ with token\$)	USPAT	OR	ON	2006/01/25 08:09
L52	3	L35 and normaliz\$	USPAT	OR	ON	2006/01/25 08:09
L53	6	L35 and normal\$	USPAT	OR	ON	2006/01/25 08:09
L54	0	(pars\$ token\$) with (anchortext (anchor adj text))	USPAT	OR	ON	2006/01/25 08:09
L55	669	(pars\$ token\$) with (URL hyperlink)	USPAT	OR	ON	2006/01/25 08:09
L56	134	(token\$) with (URL hyperlink)	USPAT	OR	ON	2006/01/25 08:09
L57	12	L56 and weight\$ and threshold\$ and normaliz\$	USPAT	OR	ON	2006/01/25 08:09
L58	58	L56 and index\$	USPAT	OR	ON	2006/01/25 08:09
L59	15	(tokeniz\$) with (URL hyperlink)	USPAT	OR	ON	2006/01/25 08:09
L60	15	(tokeniz\$) with (URL hyperlink (web adj address))	USPAT	OR	ON	2006/01/25 08:09
L61	64	(anchortext (anchor adj text))	USPAT	OR	ON	2006/01/25 08:09
L62	41	L61 and index\$	USPAT	OR	ON	2006/01/25 08:09
L63	11	L60 and index\$	USPAT	OR	ON	2006/01/25 08:09
L64	28	(index\$ categoriz\$ classif\$) near (document (web adj page) webpage) with token\$	USPAT	OR	ON	2006/01/25 08:09
L65	0	L64 same (weight and threshold)	USPAT	OR	ON	2006/01/25 08:09
L66	0	L64 same (weight\$ and threshold)	USPAT	OR	ON	2006/01/25 08:09
L67	10	L64 and (weight\$ and threshold)	USPAT	OR	ON	2006/01/25 08:09
L68	1	("5526443").PN.	US-PGPUB; USPAT; IBM_TDB	OR	OFF	2006/01/25 08:09
L69	18	L64 not L67	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:09
L70	11	weight with token same threshold	USPAT	OR	ON	2006/01/25 08:09
L71	0	8 with (substring (sub adj string))	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:14
L72	0	8 same (substring (sub adj string))	US-PGPUB; USPAT; IBM_TDB	OR	ON	2006/01/25 08:14

Terms used [anchotext tokenization](#)

Found 214 of 169,866

Sort results by relevance

Display results expanded form

 [Save results to a Binder](#)
 [Search Tips](#)
 [Open results in a new window](#)

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)


Best 200 shown

Relevance scale 

1 [One tokenization per source](#)

Jin Guo

August 1998 **Proceedings of the 36th annual meeting on Association for Computational Linguistics - Volume 1 , Proceedings of the 17th international conference on Computational linguistics - Volume 1**
Publisher: Association for Computational Linguistics , Association for Computational Linguistics

Full text available:  [pdf\(645.21 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)
 [Publisher Site](#)


We report in this paper the observation of *one tokenization per source*. That is, the same critical fragment in different sentences from the same source almost always realize one and the same of its many possible tokenizations. This observation is demonstrated very helpful in sentence tokenization practice, and is argued to be with far-reaching implications in natural language processing.


2 [Critical tokenization and its properties](#)

Jin Guo

December 1997 **Computational Linguistics**, Volume 23 Issue 4


Publisher: MIT Press

Full text available:  [pdf\(2.04 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)
 [Publisher Site](#)


Tokenization is the process of mapping sentences from character strings into strings of words. This paper sets out to study critical tokenization, a distinctive type of tokenization following the principle of maximum tokenization. The objective in this paper is to develop its mathematical description and understanding. The main results are as follows: (1) Critical points are all and only unambiguous token boundaries for any character string on a complete dictionary; (2) Any critically tokenized wo ...

3 ["Maximal-munch" tokenization in linear time](#)

 Thomas Reps

March 1998 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 20 Issue 2

Publisher: ACM Press

Full text available:  [pdf\(152.17 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The lexical-analysis (or scanning) phase of a compiler attempts to partition an input string into a sequence of tokens. The convention in most languages is that the input is scanned left to right, and each token identified is a "maximal munch" of the remaining input—the longest prefix of the remaining input that is a token of the language. Although most of the standard compiler textbooks present a way to perform maximal-munch tokenization, the algorithm th ...


Keywords: backtracking, dynamic programming, memoization, tabulation, tokenization

4 Posters: Improving Chinese tokenization with linguistic filters on statistical lexical acquisition

Dekai Wu, Pascale Fung

October 1994 **Proceedings of the fourth conference on Applied natural language processing**

Publisher: Morgan Kaufmann Publishers Inc.

Full text available:  [pdf\(250.14 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)



[Publisher Site](#)


The first step in Chinese NLP is to tokenize or segment character sequences into words, since the text contains no word delimiters. Recent heavy activity in this area has shown the biggest stumbling block to be words that are absent from the lexicon, since successful tokenizers to date have been based on dictionary lookup (e.g., Chang & Chen 1993; Chiang *et al.* 1992; Lin *et al.* 1993; Wu & Tseng 1993; Sproat *et al.* 1994). We present empirical evidence for four points concernin ...

5 Morphology, phonolgy, syntax: Tokenization as the initial phase in NLP

Jonathan J. Webster, Chunyu Kit

August 1992 **Proceedings of the 14th conference on Computational linguistics - Volume 4**

Publisher: Association for Computational Linguistics

Full text available:  [pdf\(424.19 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)


In this paper, the authors address the significance and complexity of tokenization, the beginning step of NLP. Notions of word and token are discussed and defined from the viewpoints of lexicography and pragmatic implementation, respectively. Automatic segmentation of Chinese words is presented as an illustration of tokenization. Practical approaches to identification of compound tokens in English, such as idioms, phrasal verbs and fixed expressions, are developed.

6 Optimizing encoding: An evaluation of binary xml encoding optimizations for fast stream based xml processing

R. J. Bayardo, D. Gruhl, V. Josifovski, J. Myllymaki

May 2004 **Proceedings of the 13th international conference on World Wide Web**

Publisher: ACM Press

Full text available:  [pdf\(255.72 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper provides an objective evaluation of the performance impacts of binary XML encodings, using a fast stream-based XQuery processor as our representative application. Instead of proposing one binary format and comparing it against standard XML parsers, we investigate the individual effects of several binary encoding techniques that are shared by many proposals. Our goal is to provide a deeper understanding of the performance impacts of binary XML encodings in order to clarify the ongoing ...


Keywords: XML binary formats, XPath processing

7 Language independent morphological analysis

Tatsuo Yamashita, Yuji Matsumoto

April 2000 **Proceedings of the sixth conference on Applied natural language processing**

Publisher: Morgan Kaufmann Publishers Inc.


Full text available:  [pdf\(602.81 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)




[Publisher Site](#)


This paper proposes a framework of language independent morphological analysis and mainly concentrate on tokenization, the first process of morphological analysis. Although tokenization is usually not regarded as a difficult task in most segmented languages such as English, there are a number of problems in achieving precise treatment of lexical entries. We first introduce the concept of morpho-fragments, which are intermediate units between characters and lexical entries. We describe our approa ...

- 8 Regular papers: A formalism for universal segmentation of text 
Julien Quint
July 2000 **Proceedings of the 18th conference on Computational linguistics - Volume 2**

Publisher: Association for Computational Linguistics

Full text available:  [pdf\(590.61 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)


Sumo is a formalism for universal segmentation of text. Its purpose is to provide a framework for the creation of segmentation applications. It is called "universal" as the formalism itself is independent of the language of the documents to process and independent of the levels of segmentation (e.g. words, sentences, paragraphs, morphemes...) considered by the target application. This framework relies on a layered structure representing the possible segmentations of the document. This structure ...


- 9 Toward a design apprentice: supporting reuse and evolution in software design 

Richard C. Waters, Yang Meng Tan

April 1991 **ACM SIGSOFT Software Engineering Notes**, Volume 16 Issue 2

Publisher: ACM Press



Full text available:  [pdf\(1.51 MB\)](#) Additional Information: [full citation](#), [citations](#), [index terms](#)

- 10 Special issue on computational phonology: The reconstruction engine: a computer implementation of the comparative method 


John B. Lowe, Martine Mazaudon

September 1994 **Computational Linguistics**, Volume 20 Issue 3

Publisher: MIT Press

Full text available:  [pdf\(2.14 MB\)](#)  Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)
[Publisher Site](#)


We describe the implementation of a computer program, the Reconstruction Engine (RE), which models the comparative method for establishing genetic affiliation among a group of languages. The program is a research tool designed to aid the linguist in evaluating specific hypotheses, by calculating the consequences of a set of postulated sound changes (proposed by the linguist) on complete lexicons of several languages. It divides the lexicons into a phonologically regular part and a part that devi ...

- 11 Systems: UNISYS: description of the CBAS system used for MUC-5 

Carl Weir, Rich Fritzson

August 1993 **Proceedings of the 5th conference on Message understanding MUC5 '93**

Publisher: Association for Computational Linguistics

Full text available:  [pdf\(939.34 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)


This paper describes CBAS, a data extraction system with rule-based reasoning modules. The CBAS architecture depicted in Figure 1 emphasizes the use of multiple processors to detect significant primitive facts which are then processed by reasoning modules implemented as collections of forward-chaining rules to infer additional information. A guiding principle behind the architecture is to rely as much as possible on initial processors with relatively simple internal structure in order to insure ...

- 12 An SPMD/SIMD parallel tokenizer for APL 

Robert Bernecky

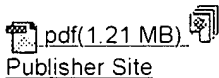
June 2003 **Proceedings of the 2003 conference on APL: stretching the mind**

Publisher: ACM Press

Full text available:  [pdf\(111.27 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)

We describe a highly parallel (SIMD within SPMD) tokenizer for the APL language, itself written in APL. The tokenizer does not break any new ground in the world of parallel computation, but does serve the didactic purpose of demonstrating that a large amount of parallelism exists in non-numeric computation. We plan to release the APEX APL Compiler, including the tokenizer, under the GNU Public License.

- 13 Dictionaries, dictionary grammars and dictionary entry parsing




We identify two complementary processes in the conversion of machine-readable dictionaries into lexical databases: recovery of the dictionary structure from the typographical markings which persist on the dictionary distribution tapes and embody the publishers' notational conventions; followed by making explicit all of the codified and ellided information packed into individual entries. We discuss notational conventions and tape formats, outline structural properties of dictionaries, observe a ra ...

14 Terminology finite-state preprocessing for computational LFG

Caroline Brun

August 1998 **Proceedings of the 36th annual meeting on Association for Computational Linguistics - Volume 1 , Proceedings of the 17th international conference on Computational linguistics - Volume 1**

Publisher: Association for Computational Linguistics , Association for Computational Linguistics

Full text available:  pdf(424.22 KB)



Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

This paper presents a technique to deal with multiword nominal terminology in a computational Lexical Functional Grammar. This method treats multiword terms as single tokens by modifying the preprocessing stage of the grammar (tokenization and morphological analysis), which consists of a cascade of two-level finite-state automata (transducers). We present here how we build the transducers to take terminology into account. We tested the method by parsing a small corpus with the without this treat ...


15 Full Technical Papers: Towards a theory of natural language interfaces to databases



Ana-Maria Popescu, Oren Etzioni, Henry Kautz

January 2003 **Proceedings of the 8th international conference on Intelligent user interfaces**

Publisher: ACM Press

Full text available:  pdf(232.68 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The need for Natural Language Interfaces to databases (NLIs) has become increasingly acute as more and more people access information through their web browsers, PDAs, and cell phones. Yet NLIs are only usable if they map natural language questions to SQL queries *correctly*. As Schneiderman and Norman have argued, people are unwilling to trade reliable and predictable user interfaces for intelligent but unreliable ones. In this paper, we introduce a theoretical framework for reliable NLIs, ...


Keywords: database, natural language interface, reliability

16 Systems: University of Pennsylvania: description of the University of Pennsylvania system used for MUC-6

Breck Baldwin, Mike Collins, Jason Eisner, Adwait Ratnaparkhi, Joseph Rosenzweig, Anoop Sarkar

November 1993 **Proceedings of the 6th conference on Message understanding MUC6 '95**

Publisher: Association for Computational Linguistics

Full text available:  pdf(1.07 MB)

Additional Information: [full citation](#), [abstract](#), [references](#)

Breck Baldwin and Jeff Reynar informally began the University of Pennsylvania's MUC-6 coreference effort in January of 1995. For the first few months, tools were built and the system was extended at weekly 'hack sessions.' As more people began attending these meetings and contributing to the project, it grew to include eight graduate students. While the effort was still informal, Mark Wasson, from Lexis-Nexis, became an advisor to the project. In July, the students proposed to the faculty that w ...

17 Document engineering (DE): Performance evaluation for text processing of noisy



inputs

Daniel Lopresti

March 2005 **Proceedings of the 2005 ACM symposium on Applied computing SAC '05**

Publisher: ACM Press

Full text available: pdf(110.60 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

We investigate the problem of evaluating the performance of text processing algorithms on inputs that contain errors as a result of optical character recognition. A new hierarchical paradigm is proposed based on approximate string matching, allowing each stage in the processing pipeline to be tested, the error effects analyzed, and possible solutions suggested.

Keywords: optical character recognition, part-of-speech tagging, performance evaluation, sentence boundary detection, tokenization

18 Design of an interpretive environment for Turing



James R. Cordy, T. C. N. Graham

July 1987 **ACM SIGPLAN Notices , Papers of the Symposium on Interpreters and interpretive techniques SIGPLAN '87**, Volume 22 Issue 7

Publisher: ACM Press

Full text available: pdf(514.77 KB) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This paper presents the design of an interpreter structure for modern programming languages such as Turing and Modula II that is modular and highly orthogonal while providing maximal flexibility and efficiency in implementation. At the outermost level, the structure consists of a front end, responsible for interaction with the user, and a back end, responsible for execution. The two are linked by a single database consisting of the tokenized statements of the user program. Interfaces between the ...

19 Multimedia: A phonotactic-semantic paradigm for automatic spoken document



classification

Bin Ma, Haizhou Li

August 2005 **Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval SIGIR '05**

Publisher: ACM Press

Full text available: pdf(311.05 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

We demonstrate a *phonotactic-semantic* paradigm for spoken document categorization. In this framework, we define a set of acoustic words instead of lexical words to represent acoustic activities in spoken languages. The strategy for acoustic vocabulary selection is studied by comparing different feature selection methods. With an appropriate acoustic vocabulary, a voice tokenizer converts a spoken document into a text-like document of acoustic words. Thus, a spoken document can be represen ...

Keywords: acoustic words, n-gram, phonotactic-semantic, semantic domain, spoken document classification, voice tokenizer

20 Database session 5: management of data streams: Raindrop: a uniform and layered



algebraic framework for XQueries on XML streams

Hong Su, Jinhui Jian, Elke A. Rundensteiner

November 2003 **Proceedings of the twelfth international conference on Information and knowledge management**

Publisher: ACM Press

Full text available: pdf(705.69 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

XML stream applications bring the challenge of efficiently processing queries on sequentially accessible token-based data. While the automata model is naturally suited for pattern matching on tokenized XML streams, the algebraic model in contrast is a well-established technique for set-oriented processing of self-contained tuples. However, neither automata nor algebraic models are well-equipped to handle both computation paradigms.

The goal of the *Raindrop* project is t ...





Keywords: XML stream, XQuery algebra, query processing

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)



token anchor text

[Homepage](#) | [Advanced Search](#)

Search using:

[Ask Jeeves](#)

[Google](#)

CUSTOM WEB FILTERS

[HotBot Skins](#) | [Preferences](#)

Date: **Before January 23 2002** [[Edit this Search](#)]

SPONSORED LINKS (filters not applied)

- [Plastic Tokens](#)

Upload your artwork or choose from our huge selection of stock designs
www.tokenfactory.com/

- [Buy Tokens Online Now](#)

Low Factory Direct Prices. Get prices and see photos online.
www.tokensdirect.com

- [Tokens - Made in U.S.A.](#)


Direct from the mint. Low prices. Stock ships same day - custom in 10
hoffmanmint.com

- [Token](#)

Products and information about **Token**.
eBay.com

- [Tokens](#)

We have what you're looking for. Tokens
www.eWoss.com

WEB RESULTS by  (Showing Results 1 - 7 of 7)

1. [Alvis Task T3.2 - Metadata Format for Enriched Documents](#)

...repeatable --> <**anchorText**>Text from this document</**anchorText** ... to use an Internet domain-name that they own, and append a locally unique...
www.miketaylor.org.uk/alvis/t3-2/m3-2.html

2. [// -*- c-basic-offset: 2 -*- /* * This file is part of the KDE](#)

...void putValue(ExecState *exec, int **token**, const Value& value, int /*attr*/); ... AnchorRev, AnchorTabIndex, AnchorTarget, **AnchorText**, AnchorBlur
www.opensource.apple.com/darwinsource/WWDC2004/WebCore-146.1/khtml/ecma/kjs

3. [/* #include "config.h"; */ /* The structure below is used to...](#)

#endif /*Wine Portions Begin*/ typedef struct tagPANOSE { U8 bFamilyType; U8 bSerifStyle; U8 bWeight; U8 bProportion; U8 bContrast;
www.woodsite.net/ports/textproc/wv/work/wv-0.7.1/wv.h

4. [#endif /*Wine Portions Begin*/ typedef struct tagPANOSE { U8](#)

#ifndef MSWORDVIEW_HEADER #define MSWORDVIEW_HEADER #ifdef __cplusplus extern "C" { #endif /* redefs of things that are either in glibc or we have to...
www.csn.ul.ie/~caolan/pub/wv/wv/wv.h

5. [#define HTML_BORDER_SET \(1<0\) // Set if -border has been configured](#)

...void Add_**token**(Pad_View &pad_view); virtual Pad_Bool Check_render ... text(Pad_String &string); virtual ~HTML_**AnchorText**(); HTML_**AnchorText**(Pad *pad
thelma.cs.unm.edu/pub/mohamad/pad_devo/generic/html.h

6. [#endif #define HTML_DEFAULT_FILL "gray90" #define HTML_DEFAULT](#)

...if (**Token** == "a") { tag = new HTML_EntryTagAnchor(tag); } else if (**Token** == "img") { tag = new HTML_EntryTagImage(tag, this); } else if (**Token**...

7. Citations: Graph structure in the Web - Broder, Kumar, Maghoul ...

...a **token** t_i to a **token** t_j in the complete reachability graph means ... approximately 3.4 million additional pages for which we had **anchortext**.

overcite.lcs.mit.edu/context/1225479/0

MORE SPONSORED LINKS

- Tokens

We have what you're looking for. Tokens

www.eWoss.com

Search for "**token anchortext**" using: Google

[Advertise](#) | [Help](#) | [Text-only Skin](#) | [Submit Site](#) | [HotBot International](#) | [Yellow Pages](#)

© Copyright 2006, Lycos, Inc. All Rights Reserved. | [Privacy Policy](#) | [Terms & Conditions](#) | [HotBot Your Site](#)
